Posudek bakalářské práce

Matematicko-fyzikální fakulta Univerzity Karlovy

Autor práce	Jakub Maroušek
Název práce	Efektivní kNN klasifikace malwaru z HTTPS dat
Rok odevzdání	2017
Studijní program	Informatika
Studijní obor	Softwarové a datové inženýrství
Autor posudku	RNDr. Leo Galamboš, Ph.D Oponent
Pracoviště	Katedra softwarového inženýrství

K celé práci

	lepší	OK	horší	nevyhovuje
Obtížnost zadání		X		
Splnění zadání		X		
Rozsah práce *... textová i implementační část, zohlednění náročnosti*	X			

Celkové zpracování považuji za solidní v rozsahu a rámci bakalářské práce. Vytknul bych chybějící (detailnější) srovnání Map-Reduce implementací používaných algoritmů a zvážení alternativních výpočetních modelů, včetně provozní náročnosti výpočetní platformy. To vše může ovlivňovat některé z měřených hodnot v experimentální části, respektive jejich rozptyl a teoretické limity.

Existují kNN implementace (např. PANDA), které jsou vytvořeny pro OpenMP/MPI a C/C++ prostředí. Nepovažuji za chybu volbu Map-Reduce a Hadoop/Java, bylo by ovšem vhodné tuto volbu teoreticky zhodnotit v textu práce s ohledem na implicitní režii zpracování.

Srovnání distribuované a jednouzlové metody je provedeno na rozdílných prostředích a HW konfiguracích, viz strana 26. Takové srovnání nemusí být zcela relevantní.

Práce obsahuje kompletní zdrojové kódy a použité skripty. Dokumentace této části by mohla být obsáhlejší. V omezeném časovém rámci, který jsem mohl věnovat čtení zdrojových kódů, se domnívám, že neobratné Java konstrukty (viz hodnocení implementační části práce) nezpůsobí zásadní neefektivitu (nevytvoří např. z lineární složitosti kvadratickou). Složitost jako takovou nejvíce ovlivňuje použitý výpočetní model, který je striktní co do možného způsobu zpracování dat, a který je v potřebném rozsahu dokumentován a odkazován z hlavního textu práce.

Textová část práce

	lepší	OK	horší	nevyhovuje
Formální úprava *... jazyková úroveň, typografická úroveň, citace*	X			
Struktura textu *... kontext, cíle, analýza, návrh, vyhodnocení, úroveň detailu*		X		
Analýza		X		
Vývojová dokumentace			X	
Uživatelská dokumentace		X		

V kontextu práce existují tři základní (implementační) problémy: jak sdílet globální datovou informaci, jak implementovat celý proces s co nejmenším počtem Map-Reduce kroků při dostatečné efektivitě, a jak zajistit rovnoměrnou distribuci do běhů. První bod řešil autor přes centrální cache, což přináší jisté nedostatky a práce to zmiňuje u jednoho z algoritmů na str. 22 (kapitola 3.3.1). Druhý bod je poněkud upozaděn - celý proces zpracování byl rozdělen na několik kroků (pivot-based metoda), přičemž se následně měří doba běhu všech těchto kroků dohromady. Každý z nich je přitom postižen režijními operacemi nosné výpočetní platformy. Třetí bod byl autorem zkoumán (strana 29), kdy narazil na problémy s nedostatkem paměti. Je otázkou, zda lze nějakým (automatickým) způsobem zajistit odolnost proti tomuto fenoménu na třídě analyzovaných algoritmů.

Text práce obsahuje drobnější nedostatky, které nesnižují srozumitelnost. Například v kapitole 2.2.1 je použita konstanta 4, jejíž možný smysl později vyjeví kapitola 2.2.2, ale v kapitole 5.1 se opět dere otázka: "proč právě čtyři?".

Matematický zápis je místy neformální, ale zároveň hutný, s definicí některých proměnných "za pochodu", např. na str. 18, což zhoršuje srozumitelnost textu. Odkaz na literaturu by měl spíše vykrývat detaily než pokrývat podstatné body. Slovní zápis algoritmu na straně 20 by bylo vhodnější podat v nějakém pseudojazyce, jak je ostatně provedeno v originálním článku Zhang et al. Slovní zápis není tak přehledný.

Výše uvedené nedostatky považuji spíše za výtky ke struktuře textu, neboť se týkají vesměs popisu algoritmů třetích stran, přičemž jsou zdroje řádně odkazovány. Protože se ale jedná o implementační práci, tak bych očekával detailnější uživatelskou a programátorskou dokumentaci.

Práce zmiňuje i některá zajímavá tvrzení, např. strana 22 *we found that directly using our test datasets would generate a single bucket*, bohužel bez bližších detailů. V tomto případě by bylo vhodné rovněž striktněji oddělit výsledky dosažené autorem a větším kolektivem.

Obrázky 5.3 a 5.4 důsledně nerozlišují k a K. V grafech 5.1 a 5.2 autor zdůvodnil atypický průběh měřených hodnot. Totéž by měl provést i v diskuzi k měřením z obrázků 5.6 a 5.8.

Implementační část práce

	lepší	OK	horší	nevyhovuje
Kvalita návrhu *... architektura, struktury a algoritmy, použité technologie*		X		
Kvalita zpracování *... jmenné konvence, formátování, komentáře, testování*		X		
Stabilita implementace	X			

Kvalita JavaDoc, zejména po formální stránce, neodpovídá obecným doporučením (velké písmeno na začátku, forma popisu parametrů, ap.).

Tato práce je implementační a obsahuje zdrojové kódy, které by si zasloužily větší pečlivost. Například v DataSplit.java je vidět velké množství atypických zápisů JavaDoc komentářů. Zdrojové kódy trpí potenciální neefektivitou, např. v GeometricGroupingBySize třídě, kde se používají nepříliš efektivní (a to nejenom v Java) konstrukty: objektové pole pro ukládání primitivních datových typů integer.

Stabilita implementace je předurčena platformou Hadoop. Běh vyžaduje zvýšení defaultních nastavení Hadoop/YARN, jinak nedobíhají úlohy ani na ukázkových datech. Container by končil s hlášením o přečerpání limitu 4.2GB virtuální RAM na hodnotě 4.7GB.

Celkové hodnocení Velmi dobře (spíše lepší)

Práci navrhuji na zvláštní ocenění Ne

Datum: 25. srpna 2017 Podpis